Calculus	Syntax	Uses
1. Arc Length	arclength(Position Vector, Variable, Lower, Upper)	Determines the arc length of a parametric equation.
2. Bound Volume	boundvol(Function 1, Function 2, Variable)	Determines the volume of the solid of revolution formed by the region bound between two curves.
3. Bound Volume (Restricted Domain)	boundvold(Function 1, Function 2, Variable, Lower Bound, Upper Bound)	Determines the volume between the solids of revolution formed by the two functions on a restricted domain.
4. Euler's Method	eulers(Differential equation, Independent variable, x0, xn, y0, step-size)	Uses euler's method to estimate the solution to a differential equation.
5. Mixing Problems	mix()	No inputs are required just press enter to start, program will ask you for inputs instead. Determines the Differential equation for mixing problems.
6. Rational Function	rational(numerator, denominator, variable)	Determines the holes, straight line asymptotes, and oblique asymptotes of the inputted rational function.
7. Surface area of a solid ⚠	**A function in terms of the axes of rotation:** surfarea(Function, Variable, Lower, Upper) **A parametric vector:** surfarea(Position Vector, Variable, Lower, Upper) **A function but NOT in terms of the axes of rotation** surfarea(Function (in terms of x), y, lower, upper)	Determines the surface area of the solid formed when rotating a function about an axes
8. Parametric 2nd Derivative ⚠	para_2nd_deriv(x(t), y(t))	Determines the 2nd derivative for a parametric equation.
9. Parametric Minimum Distance ⚠	para_closep(r(t), Point)	Determines the minimum distance between the parametric equation and an inputted point.
10. Parametric Tangent and Normal ⚠	para_tangnorm(x(t), y(t), Point)	Determines the equation of the tangent and normal to a parametric curve at an inputted point.
11. Conics ⚠	conic(Expression, Variable 1, Variable 2)	Determines the axis intercepts, centre, vertices, and any asymptotes for an inputted conic.

For Enquiries: lazymath2024@gmail.com

Complex Numbers	Syntax	Uses
1. De Moivre's Theorem	demoiv(Power, Number)	Solves the equations $z^{power} = number$
2. Circle locus first form	locicir1(Point , Radius)	Determines cartesian equation of circle in the form: $\|z - a\| = r$
3. Circle locus second form	locicir2(Point 1, Point 2, k)	Determines cartesian equation of circle in the form: $\|z - a\| = k\|z - b\|$
4. Ellipse locus	lociellp(Point 1, Point 2, Length)	Determines cartesian equation of ellipse in the form: $\|z - a\| + \|z - b\| = k$
5. Hyperbola locus	locihyp(Point 1, Point 2, Length)	Determines cartesian equation of hyperbola in the form: $\|z - a\| - \|z - b\| = k$
6. Line Locus	lociline(Point 1, Point 2)	Determines cartesian equation of line in the form: $\|z - a\| = \|z - b\|$
7. Quadratic Roots	quadroots(Number)	Determines the quadratic roots of a complex number algebraically
8. Ray	ray(Point, Angle)	Determines cartesian equation of ray
9. Segment ⚠	**Line** segment(Point 1, Point 2, Centre Radius) **Ray** segment(Point 1, Angle, Centre, Radius)	Determines the areas of the major and minor segments given the equation of a line/ray and a circle. **Note:** If a π is present in the 2nd argument, the program will interpret the equation as a ray.

Linear Algebra	Syntax	Uses
1. Line Cartesian to Vector	car2vecline(line Cartesian)	Converts equation of line from cartesian form to vector form
2. Plane Cartesian to Vector	car2vecplane(Plane Cartesian)	Converts equation of plane from cartesian form to vector form
3. Line Vector to Cartesian	vec2carline(line Vector)	Converts equation of line from vector form to cartesian form
4. Plane Vector to Cartesian	vec2carplane(Plane Vector)	Converts equation of plane from vector form to cartesian form
5. Minimum Distance between 2 lines	dist2l(Line Vector 1, Line Vector 2)	Determines minimum distance between two lines
6. Minimum Distance between 2 planes	dist2pl(Plane Cartesian 1, Plane Cartesian 2)	Determines minimum distance between two planes
7. Minimum Distance between line and plane	distlpl(Line Vector, Plane Cartesian)	Determines the minimum distance between a plane and line
8. Minimum Distance between line and point	distlp(Line Vector, Point)	Determines minimum distance between a line and point
9. Minimum Distance between plane and point	distlp(Plane Equation, Point)	Determines minimum distance between a plane and point
10. Intersection between 2 lines	ints2l(Line Vector 1, Line Vector 2)	Determines the point of **intersection** & **angle** between two lines
11. Intersection between 2 planes	ints2pl(Plane Cartesian 1, Plane Cartesian 2)	Determines the line of **intersection** & **angle** between two planes
12. Intersection between plane and line	intslpl(Line Vector, Plane Cartesian)	Determines the point of **intersection** & **angle** between line and plane
13. Create line with 2 points	line2p(Point 1, Point 2)	Determines the equation of a line given two points
14. Create line with direction vector and point	linedp(Direction Vector, Point)	Determines the equation of a line given a direction vector and point
15. Create plane with 3 points	plane3p(Point 1, Point 2, Point 3)	Determines the equation of a plane given three points
16. Create plane with normal and point	planenp(Normal Vector, Point)	Determines the equation of a plane given a normal vector and a point
17. Plane formed by intersecting lines	planeintl(Line Vector 1, Line Vector 2)	Determines the equation of the plane formed by two intersecting lines

For Enquiries: lazymath2024@gmail.com

Vectors	Syntax	Uses
1. Unit Vector Bisector	bisec(vector 1, vector 2)	Determines the unit vector which bisects the angle between two vectors
2. Colinear	colin(Point 1, Point 2, Point 3)	Determines value(s) of a variable required for points to be collinear
3. Linear Dependence	lindep(Vector 1, Vector 2, Vector 3)	Determines value(s) of a variable required for 3 vectors to be linearly dependent
4. Vector Projection	vproj(Vector 1, Vector 2)	Determines vector, scalar resolute, & angle for two inputted vectors
5. Angle between Two Vectors	vecang(Vector 1, Vector 2)	Determines the angle between two inputted vectors.

Kinematics	Syntax	Uses
1. Collision Detector	collision(Position Vector 1, Position Vector 2)	Determines whether two particles collide and the points at which their paths intersect.
2. Projectile Motion	projm(Initial Position, Initial Velocity, Angle of launch to horizontal (in degrees), Initial Acceleration) For example, Bob throws a ball at **20m/s** at an angle of **30 degrees to the horizontal**. The ball accelerates downwards due to **gravity**. Bob on a **60m** high cliff. for example, projm(60, 20, 30, -49/5)	Determines the maximum height, horizontal displacement, and speed as projectile hits the ground as well as all the acceleration, velocity and position vectors.
3. suvat	suvat(s,u,v,a,t)	Give 2 variables and 3 known values, it does the rest.
4. kin_desolve ⚠	kin_desolve(acceleration, variable 1, variable 2, initial condition 1, initial condition 2)	Solves an acceleration differential equation. You must provide it with the variables and their corresponding initial condition.

Probability and Statistics	Syntax	Uses
1. Sample Mean Confidence interval	confint(Sample mean, population standard deviation, sample size, . confidence)	Determines confidence interval for sample mean.
2. Hypothesis testing	hyptest()	Performs either a one tailed or two tailed hypothesis test. You will be prompted for the inputs.
3. Error probability	prerror()	Determines the probability of either a type I or type II error occurring. You'll be prompted for the inputs.
4. P-value	pval(sample mean, population mean, population standard deviation, sample size)	Determines p-values of both one and two tailed tests.

For Enquiries: lazymath2024@gmail.com